

# JavaScript, programmation avancée

3 j (21 heures)

Ref : JSAV

## Public

Développeurs Web, architectes web, chefs de projet, webmasters...

## Pré-requis

Avoir des connaissances pratiques en JavaScript, HTML et CSS. Posséder des connaissances de base de XML.

## Moyens pédagogiques

Modalité : Formation présentielle ou Formation distancielle (classe virtuelle) - Inter / Intra - Groupes de 4 à 12 stagiaires

Méthodes : Présentation des concepts, discussion technique, démonstrations, exercices et TP

Matériel :

Présentiel : Un poste informatique par stagiaire connecté à internet, à une imprimante en réseau et au réseau informatique,

Les salles sont équipées d'un tableau interactif ou d'un vidéoprojecteur et d'un paperboard

Distanciel : Aelion met à disposition de chaque stagiaire

- Un PC équipé des outils et logiciels nécessaires à la formation auquel vous accédez via un outil de prise en main à distance

- Un accès à un outil de classe virtuelle (Meet)

Support de formation : Un support de formation sera remis à chaque stagiaire en fin de formation : plateforme collaborative intégrant le code source des exercices réalisés en formation, webographie, mémos

## Modalités de suivi et d'évaluation

Modalités de suivi et d'évaluation

Questionnaire d'évaluation des pré-requis, suivi des connaissances tout au long de la formation, Evaluation des acquis en fin de formation

Questionnaire d'évaluation de la satisfaction en fin de stage, feuille de présence émarginée par demi-journée par les stagiaires et le formateur, Attestation de fin de formation

Longtemps relégué dans la catégorie des simples langages de script, JS est aujourd'hui un acteur incontournable dans le développement web et web mobile. Tant sur la pile frontend que backend, JS tire partie aujourd'hui d'un socle solide EcmaScript, d'une syntaxe plus proche des langages « conventionnels » objet et d'un nombre de modules et frameworks de plus en plus important pour couvrir l'ensemble des besoins en matière de développement web. Au cours de cette formation, vous pourrez découvrir les spécificités du langage avec l'introduction d'ES, d'approcher la programmation fonctionnelle, de découvrir la programmation JS côté serveur avec NodeJS et Express ainsi que les outils de build (npm, webpack, ...). A l'issue de la formation, vous serez capable de développer des applications client riches à partir de concepts avancés JavaScript, en mettant en oeuvre les normes EcmaScript et en mobilisant les bons outils et frameworks associés.

**Action collective OPCO ATLAS - [Inscription CampusAtlas](#)**

## Objectifs

- Découvrir les concepts avancés JavaScript ES5
- Maîtriser l'environnement de débogage
- Manipuler les API JavaScript HTML5
- Mettre en pratique la Programmation Orientée Objet avec les Web Component
- Mettre en oeuvre le framework JavaScript jQuery
- Appréhender la notion de JavaScript côté serveur avec Node.js

## Programme détaillé

### DECOUVRIR LES CONCEPTS AVANCES JAVASCRIPT ES6

---

A l'issue de cette séquence, vous êtes capable d'écrire du code Javascript en utilisant le standard ES6. Vous comprenez la notion de module. Vous prenez conscience des outils liés à la rétro-compatibilité.

Les syntaxes fondamentales ES6

- Variables avec let et const
- Les classes
- Fonctions fléchées
- Import et export de module statique

Autres nouveautés et rétrocompatibilité (compilers, polyfills)

### MAITRISER L'ENVIRONNEMENT ET LE DEBOGAGE

---

A l'issue de cette séquence, vous êtes capable de mettre en place un atelier de génie logiciel pour Javascript et d'utiliser les outils de build/dépendances, et de débogage.

Gestionnaires de builds et dépendances

- Npm et npm install
- Le cycle de vie de npm
- Utilisation des scripts npm

Les outils de haut niveau webpack, babel

Outils de débogage

### ÉCRIRE ET UTILISER DU CODE JAVASCRIPT

---

A l'issue de cette séquence, vous êtes capable d'écrire du code javascript moderne avec les syntaxes du standard ES6 ou plus.

Principes et nouveautés de ES6 et plus

- Syntaxes de bases d'écriture du langage
- Mots-clés (let, const)
- Fonctions
- Les fonctions fléchées
- Les fonctions de manipulation de données : map, filter, forEach...

La programmation orientée objet en ES6

- Définition des classes

- L'opérateur new
  - Modification d'un objet : getter / setter
  - Héritage
- Rétrocompatibilité
- Table des compatibilités
  - Polyfills
  - Compilateurs

## **MANIPULER LES API JAVASCRIPT HTML5 AVANCEES**

---

A l'issue de cette séquence, vous êtes capable d'utiliser des technologies de rendu moderne grâce à JS. Vous savez utiliser les technologies de stockage et les traitements en arrière-plan côté client.

Les technologies de rendu côté client

- Graphisme avec Canvas
- La 3D avec WebGL et Three.js
- La datavisualisation avec D3.js

La donnée et les traitements côté client

- Le stockage local : sessionStorage, localStorage, indexedDB
- Les traitements en arrière-plan: Web Workers, Service Workers

## **METTRE EN PRATIQUE LA PROGRAMMATION ORIENTEE OBJET AVEC LES WEB COMPONENT**

---

A l'issue de cette séquence, vous êtes capable de créer des composants. Vous savez isoler ses composants sous forme de module (ES modules) et les charger pour les utiliser dans une page.

La notion de Single Page App (SPA)

Les problématiques de structure des applications HTML modernes avec les SPA

L'utilisation de l'API Custom Elements pour créer des composants réutilisables

## **METTRE EN ŒUVRE LE FRAMEWORK JAVASCRIPT JQUERY**

---

A l'issue de cette séquence, vous êtes capable d'identifier les apports des librairies et des Framework Javascript. Vous savez identifier les librairies et Framework les plus connus.

Les limites du JavaScript natif

Identifier et utiliser des librairies utiles

Identifier le rôle des principaux Framework

## **APPREHENDER LA NOTION DE JAVASCRIPT COTE SERVEUR AVEC NODE.JS**

---

A l'issue de cette séquence, vous êtes capable d'identifier le rôle de Node.js et son utilisation pour créer des applications côté serveur (Backend).

Introduction à Node.js

L'environnement Node.js

Le temps réel avec les WebSockets et la librairie Socket.io